Name: Sahib Bajwa
ID: 107553096

CSCI 3104, Algorithms | **Profs. Chen & Grochow**
Problem Set 2 – Due Thurs Jan 30 11:55pm | **Spring 2020, CU-Boulder**

Advice 1: For every problem in this class, you must justify your answer: show how you arrived at it and why it is correct. If there are assumptions you need to make along the way, state those clearly.

Advice 2: Informal reasoning is typically insufficient for full credit. Instead, write a logical argument, in the style of a mathematical proof.

Instructions for submitting your solutions:

- The solutions **should be typed** and we cannot accept hand-written solutions. Here's a short intro to LaTeX.
- You should submit your work through the **class Canvas page** only.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this template of at least 9 pages (or Gradescope has issues with it).

Quicklinks: 1 2a 2b 2c 2d 3a 3b 3c

---

1. *Name (a) one advantage, (b) one disadvantage, and (c) one alternative to worst-case analysis. For (a) and (b) you should use full sentences.*

Homework 2

1. a) Tells us the longest running time for any input of size n.

b) Lower bounds tell us nothing about worst case scenarios. Worst case may be average case.

c) Best case analysis

Name: Sahib Bajwa
ID: 107553096

2. *For each part of this question, put the growth rates in order, from slowest-growing to fastest. That is, if your answer is $f_1(n), f_2(n), \ldots, f_k(n)$, then $f_i(n) \leq O(f_{i+1}(n))$ for all i. If two adjacent ones are asymptotically the same (that is, $f_i(n) = \Theta(f_{i+1}(n))$), you must specify this as well.*

   Justify your answer (show your work). You may assume transitivity: if $f(n) \leq O(g(n))$ and $g(n) \leq O(h(n))$, then $f(n) \leq O(h(n))$, and similarly for little-oh, etc.

   (a) *Polynomials.*

$$n+1 \qquad n^4 \qquad 1/n \qquad 1 \qquad n^2+2n-4 \qquad n^2 \qquad \sqrt{n} \qquad 10^{100}$$

2.a) $\lim_{n\to\infty} \frac{n+1}{n^4} \Rightarrow \lim_{n\to\infty} \frac{1}{n^3} \to 0$ $\qquad n+1 < n^4$

$\lim_{n\to\infty} \frac{\frac{1}{n}\downarrow}{n^4\uparrow} \to 0 \qquad \lim_{n\to\infty} \frac{\frac{1}{n}}{n+1} \to \frac{\frac{1}{n}\downarrow}{n\uparrow} \to 0$ $\qquad \frac{1}{n} < n+1 < n^4$

$\lim_{n\to\infty} \frac{\frac{1}{n}\downarrow}{1} \to 0 \qquad \lim_{n\to\infty} \frac{n+1\uparrow}{1\downarrow} \to \infty$ $\qquad \frac{1}{n} < 1 < n+1 < n^4$

$\lim_{n\to\infty} \frac{n^4}{n^2+2n-4} \to \frac{n^4}{n^2+2n} \to \frac{n^3}{n+2} \to \frac{n^2\uparrow}{1} \to \infty$

$\lim_{n\to\infty} \frac{n^2+2n-4}{n+1} \to \frac{n^2+2n}{n} \to \frac{n+2\uparrow}{1} \to \infty$ $\qquad \frac{1}{n} < 1 < n+1 < n^2+2n-4 < n^4$

$\lim_{n\to\infty} \frac{n^2+2n-4}{n^2} \to \lim_{n\to\infty} \frac{n^2-2n}{n^2} \to \lim_{n\to\infty} \frac{n^2}{n^2} = 1$ $\qquad \frac{1}{n} < 1 < n+1 < n^2 = n^2+2n-4 < n^4$

$\lim_{n\to\infty} \frac{\sqrt{n}}{n+1} \xrightarrow{L'Hop} \frac{\frac{1}{2\sqrt{n}}}{1} \to 0$

$\lim_{n\to\infty} \frac{\sqrt{n}}{1} \longrightarrow \infty$ $\qquad \frac{1}{n} < 1 < \sqrt{n} < n+1 < n^2 = n^2+2n-4 < n^4$

$\lim_{n\to\infty} \frac{10^{100}}{1} = 10^{100}$ $\qquad \boxed{\frac{1}{n} < 1 = 10^{100} < \sqrt{n} < n+1 < n^2 = n^2+2n-4 < n^4}$

(b) *Logarithms and related functions.*

$$(\log_2 n)^2 \qquad \log_2(n) \qquad \log_3(n) \qquad \sqrt{n} \qquad \log_{1.5}(n) \qquad \log_2(n^2)$$

b)

$$\lim_{n\to\infty} \frac{(\log_2 n)^2}{\log_2 n} \xrightarrow{L'H} \frac{\frac{2\ln n}{(\ln 2)^2 n}}{\frac{1}{(\ln 2)}} \xrightarrow{\text{simplify}} \frac{2\ln(n)\cdot \cancel{\ln(2)}\cdot \cancel{n}}{(\ln(2))(\ln(2))\cdot\cancel{x}} \xrightarrow{\text{simplify}} \frac{2\ln(n)\cdot \cancel{n}}{\ln(2)} \to \infty$$

$$\log_2(n) < (\log_2 n)^2$$

$$\lim_{n\to\infty} \frac{\log_3 n}{\log_2 n} \xrightarrow{L'H} \frac{\frac{1}{\ln(3)\cdot x}}{\frac{1}{\ln(2)\cdot x}} \xrightarrow{\text{simplify}} \frac{\ln(2)\cdot\cancel{x}}{\ln(3)\cdot\cancel{x}} \xrightarrow{\text{simplify}} \frac{\ln(2)}{\ln(3)} = .631$$

$$\log_2(n) = \log_3(n) < (\log_2 n)^2$$

$$\lim_{n\to\infty} \frac{\sqrt{n}}{(\log_2 n)^2} \xrightarrow{L'H} \frac{\frac{1}{2\sqrt{n}}}{\frac{2\ln(n)}{(\ln(2))^2\cdot n}} \to \frac{\ln(2)^2\cdot n}{2\sqrt{n}\cdot 2\ln(n)} \xrightarrow{\text{simplify}} \frac{n}{\sqrt{n}\cdot\ln(n)} \xrightarrow{L'H} \frac{1}{\frac{\ln(n)+2}{2\sqrt{n}}} \xrightarrow{\text{simplify}} \frac{\sqrt{n}}{\ln(n)} \xrightarrow{L'H}$$

$$\frac{\frac{1}{2\sqrt{n}}}{\frac{1}{n}} \xrightarrow{\text{simplify}} \frac{n}{\sqrt{n}} \xrightarrow{\text{simplify}} \frac{1}{\frac{1}{2\sqrt{n}}} \xrightarrow{\text{simplify}} \frac{2\sqrt{n}}{1} \to \infty$$

$$\log_2(n) = \log_3(n) < (\log_2(n))^2 < \sqrt{n}$$

$$\lim_{n\to\infty} \frac{(\log_2(n))^2}{\log_{1.5}(n)} \xrightarrow{L'H} \frac{\frac{2\ln(n)}{(\ln(2))^2\cdot n}}{\frac{1}{\ln(\frac{3}{2})\cdot n}} \xrightarrow{\text{simplify}} \frac{\frac{\ln(n)}{n}}{\frac{1}{n}} \xrightarrow{\text{simplify}} \frac{\ln(n)\cdot\cancel{n}}{\cancel{n}} \xrightarrow{\text{simplify}} \frac{\ln(n)}{1} \to \infty$$

$$\lim_{n\to\infty} \frac{\log_{1.5}(n)}{\log_3(n)} \xrightarrow{L'H} \frac{\frac{1}{\ln(\frac{3}{2})n}}{\frac{1}{\ln(3)n}} \xrightarrow{\text{simplify}} \frac{\ln(3)n}{\ln(\frac{3}{2})n} \xrightarrow{\text{simplify}} \frac{n}{n} \to 1$$

$$\log_2(n) = \log_3(n) = \log_{1.5}(n) < (\log_2(n))^2 < \sqrt{n}$$

$$\lim_{n\to\infty} \frac{(\log_2(n))^2}{\log_2(n^2)} \xrightarrow{L'H} \frac{\frac{2\ln(n)}{\ln(2)^2 n}}{\frac{2}{\ln(2)n}} \xrightarrow{\text{simplify}} \frac{\cancel{\ln(2)}\, n\cdot\cancel{2}\ln(n)}{\cancel{\ln(2)^2}\cdot n\cdot\cancel{2}} \xrightarrow{\text{simplify}} \frac{\cancel{n}\ln(n)}{\cancel{n}} \xrightarrow{\text{simplify}} \frac{\ln(n)}{1} \to \infty$$

$$\lim_{n\to\infty} \frac{\log_2(n^2)}{\log_{1.5}(n)} \xrightarrow{L'H} \frac{\frac{2}{\ln(2)n}}{\frac{1}{\ln(\frac{3}{2})\cdot n}} \xrightarrow{\text{simplify}} \frac{\cancel{\ln(\frac{3}{2})}n\cdot 2}{\cancel{\ln(2)}\, n} \xrightarrow{\text{simplify}} \frac{n}{n} \to 1$$

$$\boxed{\log_2(n) = \log_3(n) = \log_{1.5}(n) < (\log_2(n))^2 < \sqrt{n}}$$

Name: Sahib Bajwa
ID: 107553096

(c) *Logarithms in exponents.*

$$n^{\log_3(n)} \qquad n^{\log_2 n} \qquad n^{1/\log_2(n)} \qquad n \qquad 1$$

c) $\lim_{n\to\infty} \frac{\log_3(n)}{\log_2(n)} \xrightarrow{L'H} \frac{\frac{1}{\ln(3)\cdot n}}{\frac{1}{\ln(2)\cdot n}} \xrightarrow{\text{simplify}} \frac{\ln(2)\cdot \cancel{n}}{\ln(3)\cdot \cancel{n}} \xrightarrow{\text{simplify}} \frac{\ln(2)}{\ln(3)}$

Since their exponent's growth rates are the same, so are $n^{\log_3(n)}$ and $n^{\log_2(n)}$'s growth rates.
$\underline{n^{\log_3(n)} = n^{\log_2(n)}}$

$\lim_{n\to\infty} \frac{\log_3(n)}{\frac{1}{\log_2(n)}} \xrightarrow{\text{simplify}} \frac{\log_3(n)\cdot\log_2(n)}{1} \to \infty$ $\quad \underline{n^{1/\log_2(n)} < n^{\log_3(n)} = n^{\log_2(n)}}$

$\lim_{n\to\infty} \frac{n^{\log_3(n)}}{n} \xrightarrow{L'H} \frac{\left(2n^{\frac{\ln(n)}{\ln(3)}-1}\cdot\ln(n)\right)/\ln(3)}{1} \to \infty$

$n^{1/\log_2(n)}$ is always 2 for any input $x \to \infty$

$\lim_{n\to\infty} \frac{n}{n^{1/\log_2(n)}} \xrightarrow{\text{substitute}} \lim_{n\to\infty} \frac{n}{2} \to \infty$ $\quad \underline{n^{1/\log_2(n)} < n < n^{\log_3(n)} = n^{\log_2(n)}}$

$\lim_{n\to\infty} \frac{1}{n^{1/\log_2(n)}} \xrightarrow{\text{substitute}} \lim_{n\to\infty} \frac{1}{2} \to \frac{1}{2}$ $\quad \underline{1 = n^{1/\log_2(n)} < n < n^{\log_3(n)} = n^{\log_2(n)}}$

$$\boxed{1 = n^{1/\log_2(n)} < n < n^{\log_3(n)} = n^{\log_2(n)}}$$

Name: Sahib Bajwa
ID: 107553096

(d) *Exponentials.* Hint: *Recall Stirling's approximation, which says that* $n! \sim \left(\frac{n}{e}\right)^n \sqrt{2\pi n}$, *i.e.* $\lim_{n\to\infty} \frac{n!}{\left(\frac{n}{e}\right)^n \sqrt{2\pi n}} = 1$.

$$n! \qquad 2^n \qquad 2^{2n} \qquad 2^{n\log_2(n)} \qquad 2^{n+7}$$

d) $\lim_{n\to\infty} \frac{2^n}{n!}$ Squeeze Theorem! $\frac{2^n}{n!} = \frac{2}{1}\cdot\frac{2}{2}\cdot\frac{2}{3}\cdot\frac{2}{4}\cdots\frac{2}{n} \le \frac{2}{1}\cdot\frac{2}{2}\cdot\frac{2}{3}\cdot\frac{2}{3}\cdot\frac{2}{3}\cdots\frac{2}{3} =$

$= \frac{2}{1}\cdot\frac{2}{2}\cdot\left(\frac{2}{3}\right)^{n-2} = \frac{4}{2}\cdot\left(\frac{2}{3}\right)^{n-2}$

$\lim_{n\to\infty} 2\left(\frac{2}{3}\right)^{n-2} \xrightarrow{\text{solve}} 0$ $\qquad$ $\underline{2^n < n!}$

$\lim_{n\to\infty} \frac{2^{2n}}{n!} \longrightarrow \frac{2^{2n}}{n!} = \frac{4}{1}\cdot\frac{4}{2}\cdot\frac{4}{3}\cdot\frac{4}{4}\cdots\frac{4}{n} \le \frac{4}{1}\cdot\frac{4}{2}\cdot\frac{4}{3}\cdot\frac{4}{3}\cdot\frac{4}{3}\cdots\frac{4}{3} =$

$= \frac{16}{2}\cdot\left(\frac{4}{3}\right)^{n-2}$

$\lim_{n\to\infty} 8\cdot\left(\frac{4}{3}\right)^{n-2} \to 0$

$\lim_{n\to\infty} \frac{2^{2n}}{2^n} \xrightarrow{\text{simplify}} \frac{2^n}{1} \to \infty$ $\qquad$ $\underline{2^n < 2^{2n} < n!}$

$\lim \frac{2^{n\log_2(n)}}{n!} \xrightarrow{\text{simplify}} \frac{n^n}{n!} \longrightarrow \frac{n^n}{n!} = \frac{1^1}{1}\cdot\frac{2^2}{2}\cdot\frac{3^3}{3}\cdots\frac{n^n}{n} \le \frac{1^1}{1}\cdot\frac{2^2}{2}\cdot\frac{3^3}{3}\cdot\frac{4^4}{3}\cdot$

$\frac{5^5}{3}\cdot\frac{6^6}{3}\cdots\frac{n^n}{3} = \frac{1^1}{1}\cdot\frac{2^2}{2}\cdot\left(\frac{n^n}{3}\right)^{n-2}$

$\lim_{n\to\infty} 2\left(\frac{n^n}{3}\right)^{n-2} \to \infty$ $\qquad$ $\underline{2^n < 2^{2n} < n! < 2^{n\log_2(n)}}$

$\lim_{n\to\infty} \frac{2^{n+7}}{2^{2n}} \xrightarrow{\text{simplify}} \lim_{n\to\infty} 2^{-n+7} \to 0$

$\lim_{n\to\infty} \frac{2^{n+\cancel{7}}}{2^n} \xrightarrow{\text{simplify}} \lim_{n\to\infty} \frac{2^n}{2^n} \to 1$ $\qquad$ $\boxed{2^n = 2^{n+7} < 2^{2n} < n! < 2^{n\log_2(n)}}$

Name: Sahib Bajwa
ID: 107553096

CSCI 3104, Algorithms — **Profs. Chen & Grochow**
Problem Set 2 – Due Thurs Jan 30 11:55pm — **Spring 2020, CU-Boulder**

3. *For each of the following algorithms, analyze the worst-case running time. You should give your answer in big-Oh notation. You* do not *need to give an input which achieves your worst-case bound, but you should try to give as tight a bound as possible.*

   Justify your answer (show your work). This likely means discussing the number of atomic operations in each line, and how many times it runs, writing out a formal summation for the runtime complexity $T(n)$ of each algorithm, and then simplifying your summation.

   (a)
```
1  f(A): // A is a square, 2D array; indexed starting from 1
2    let d be a copy of A
3    for i = 1 to len(A):
4      d[i][i] = 0
5
6    for i = 1 to len(A):
7      for j = 1 to len(A):
8        for k = 1 to len(A):
9          if (d[i][k] + d[k][j]) < d[i][j]:
10           d[i][j] = d[i][k] + d[k][j]
11
12   return d
```

3.a

	cost	times
1	0	–
2	0	–
3	c_3	n
4	c_4	$n-1$
5	0	–
6	c_6	n
7	c_7	n^2
8	c_8	n^3
9	c_9	$n-1$
10	c_{10}	$n-1$
11	0	
12	0	

$$T(n) = c_3 n + c_4(n-1) + c_6 n + c_7 n^2 + c_8 n^3 + c_9(n-1) + c_{10}(n-1)$$

$$\Rightarrow$$

$$\boxed{T(n) = \Theta(n^3)}$$

If it is blurry to see answer, worst case is n^3

(b)
```
1 g(A): // A is a list of integers
2   for i = 1 to len(A):
3     for j = 1 to len(A)-i:
4       if A[j+1] > A[j]:
5         // swap A[j+1] and A[j]
6         tmp = A[j+1]
7         A[j+1] = A[j]
8         A[j] = tmp
9   return A
```

b)

	cost	times
1	0	—
2	c_2	n
3	c_3	n^2
4	c_4	$n-1$
5	0	—
6	c_6	$n-1$
7	c_7	$n-1$
8	c_8	$n-1$
9	0	—

$$T(n) = c_2 n + c_3 n^2 + c_4(n-1) + c_6(n-1) + c_7(n-1) + c_8(n-1)$$

$\Rightarrow$

$$\boxed{T(n) = \Theta(n^2)}$$

If it is blurry to see answer, worst case is n^2

(c) Here, `abs(n)` returns the absolute value of its argument, and can be treated as an atomic operation

```
1 h(A): // A is a list of integers, of length at least 2, first index is 1
2   min = abs(A[1] - A[2])
3   for i = 1 to len(A):
4     for j = i+1 to len(A):
5       if abs(A[i] - A[j]) < min:
6         min = abs(A[i] - A[j])
7   return min
```

c)

	cost	times
1	0	–
2	c_2	1
3	c_3	n
4	c_4	n^2
5	c_5	$n-1$
6	c_6	$n-1$
7	0	–

$T(n) = c_2 + c_3 n + c_4 n^2 + c_5(n-1) + c_6(n-1)$

$\Rightarrow$

$\boxed{T(n) = \Theta(n^2)}$

If it is blurry to see answer, worst case is n^2